How to Create a New *RadExPro* Processing Project and Load Input Data
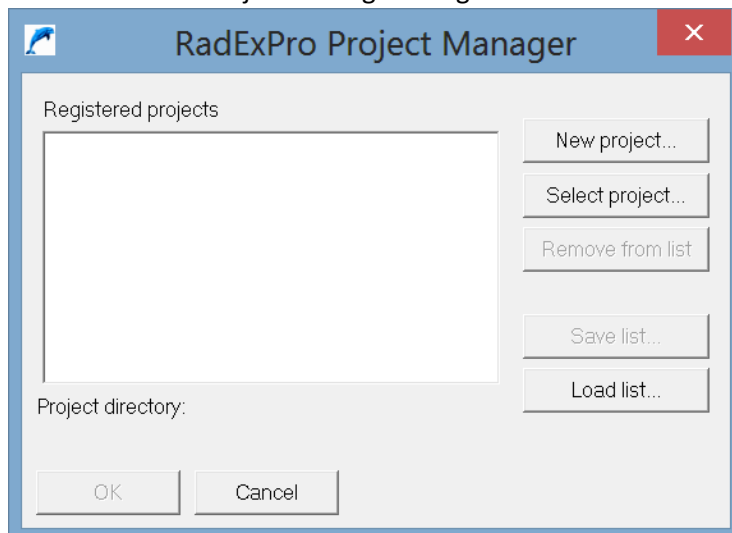
All processing in RadExPro is performed within processing *projects*. A project is a database containing raw seismic data, intermediate and final processing results together with all processing flows and parameters, velocity tables, horizon picks and auxiliary information. Each project and all of its files are stored in one folder on your hard disk. Before you can start processing your data in RadExPro you need to load it to a processing project.

Below, on the steps 1-7 we describe how you create a new RadExPro processing project.

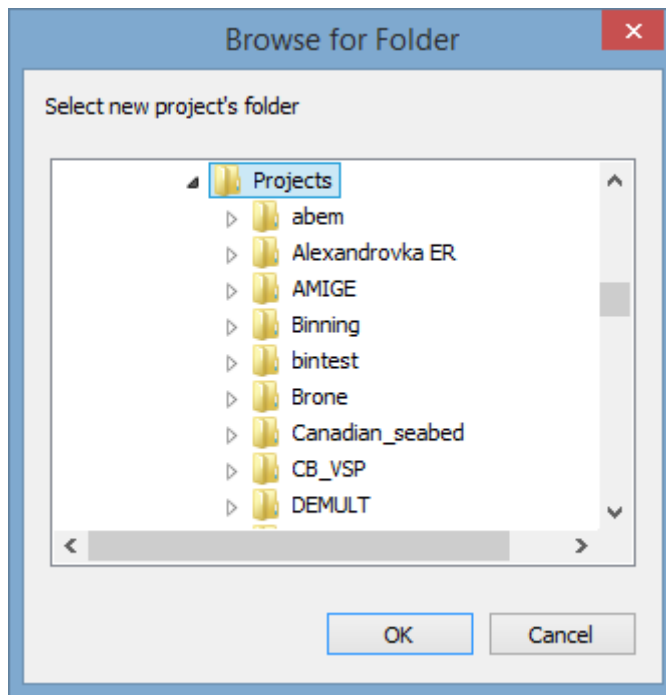Then, steps 8-14 describe loading input data into the project.

Creating a new project

1. Run the RadExPro. In previous versions of Windows you will find it in the Start menu, All programs/DECO Geophysical, in Windows 8 you need to go to the Start screen/All applications/DECO Geophysical.

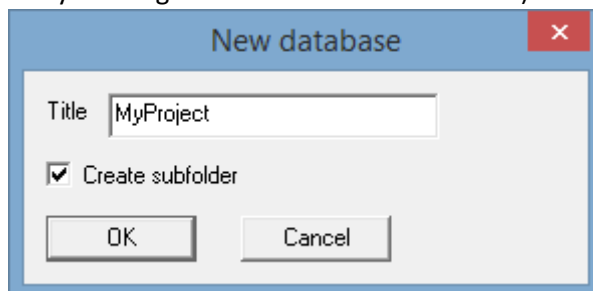2. You will see the Project Manage dialog window:

If you run the software for the first time, the list of registered projects will be empty.

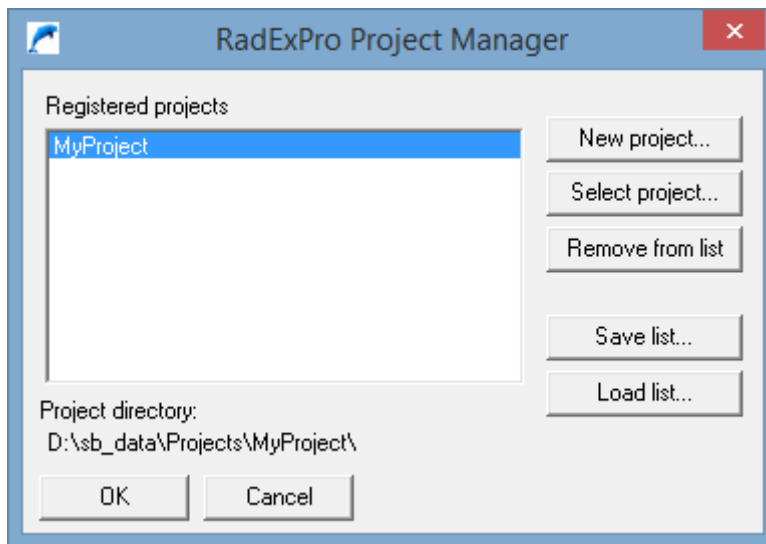3. Click the New project button to see standard Select Folder dialog:

Select a folder where your new project will be created (in this example we select a folder called 'Projects') and click the OK button.

4. In the dialog that opens type a name of the project to be created (here, we call it 'MyProject' but you can give it a better name of course):
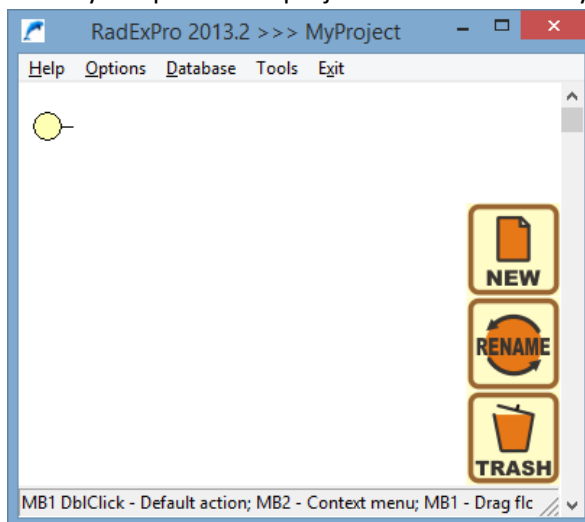

Note the Create subfolder flag: when it is checked (default), a new project sub-folder will be created inside the folder you have selected at the previous step. That is in our case, inside the 'Projects' folder a new sub-folder 'MyProject' will be created where the new project database files will be located in. When the flag is unchecked, the new project database files will be placed directly inside the 'Projects' folder.

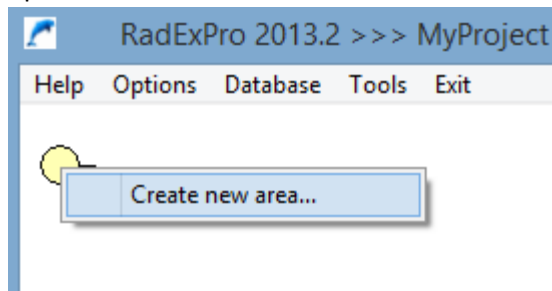5. Now the newly created project will appear in the Project Manager List.

Double-click on the project name to open it in the RadExPro.

6. When you open a new project for the first time you will see a window like this:
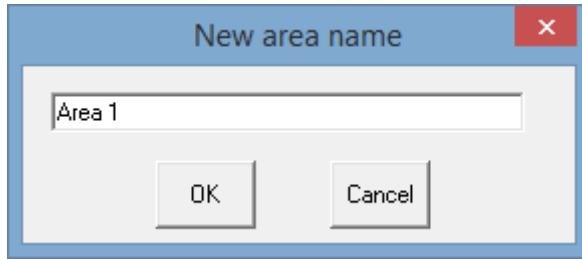

This is the main RadExPro window with the project tree, which, so far, is empty – the yellow circle in the top-left is the *root* of the project database.

7. A RadExPro project database is a 3-level structure. We call these levels as '*areas*', '*lines*' and '*processing flows*'. Each project can contain several areas (although most of the people, in fact, prefer to keep one area per project). Each line can contain a number of lines, and finally each line will contain some flows. Before we can load the data, let us create some basic project structure – an area, a line and a flow:

    a. Right-click on the root yellow circle and in the pop-up menu select Create new area… option:
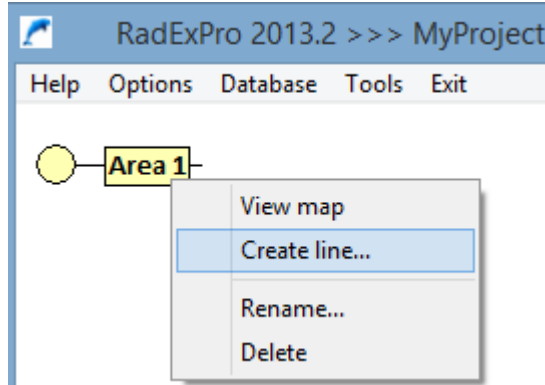
We will call our area 'Area 1' (of course you are encouraged to give it some meaningful name instead) and click the OK button.
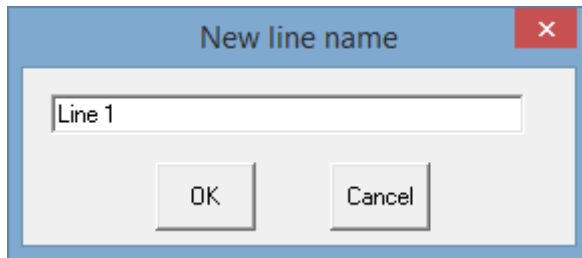

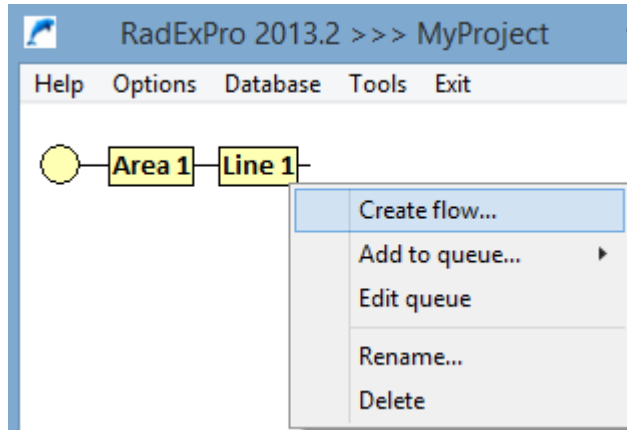You will see the area just created in the project tree.

b.   Now, we shall create a line. Right-click on the area in and select Create line… command in the pop-up menu:
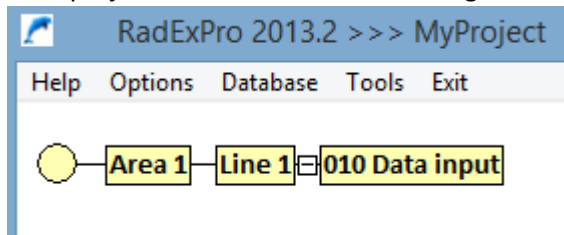

We will call our line 'Line 1'.

c.   The same way, we right-click on the line to create our first processing flow:

We are going to use this flow to load input data into the project so will call it '010 Data input' (the software will sort the flows inside a line alphabetically, so we recommend to start flows names with numbers to keep them is order).
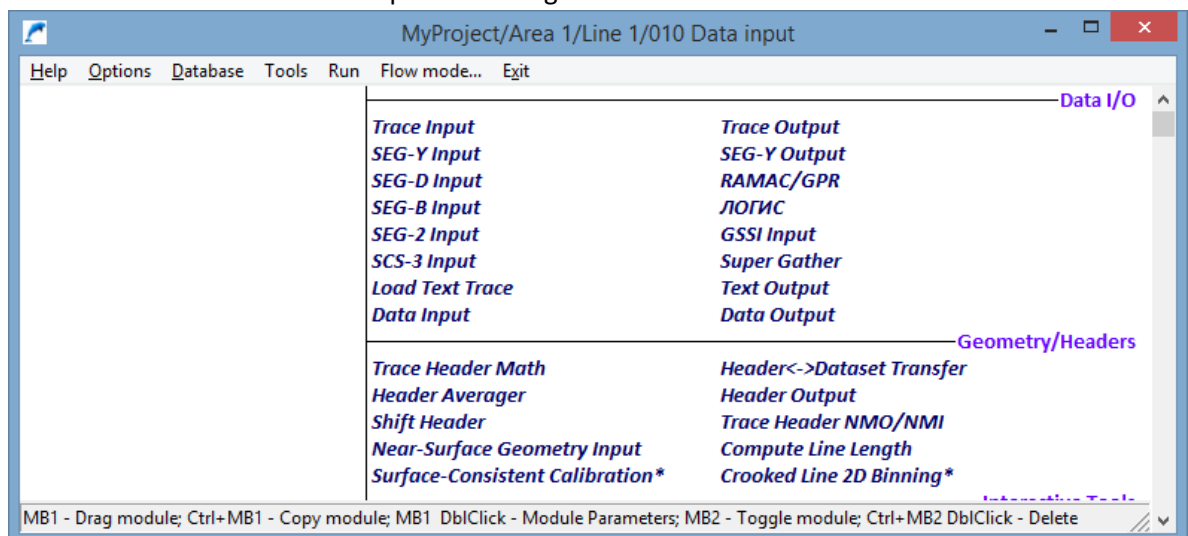
Your project tree shall look as following:

Now we are ready to load some data into the project.
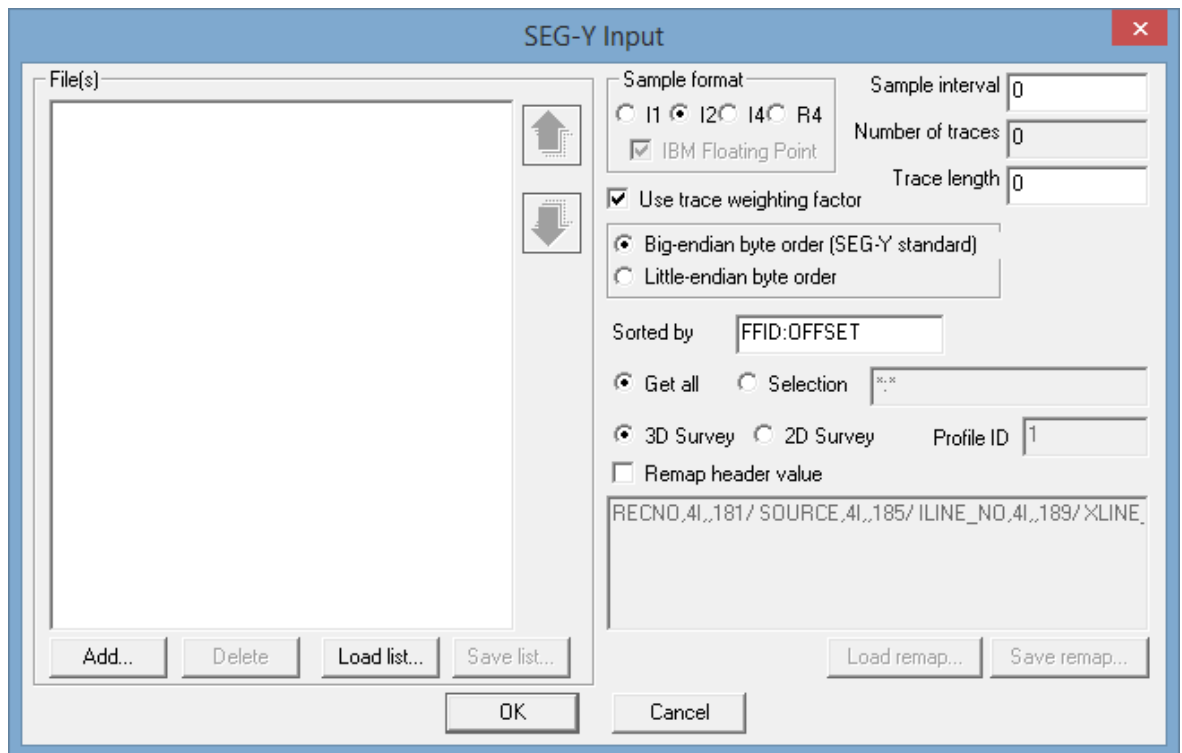
Loading data

8. We will create a sub-folder called 'DATA' inside of the project folder and copy our raw data files there. Here for this example we have got several near-surface reflection shots, each one in a separate SEG-Y file.

9. Double-click on the '010 Data input' flow[1] to get into the Flow Editor window:
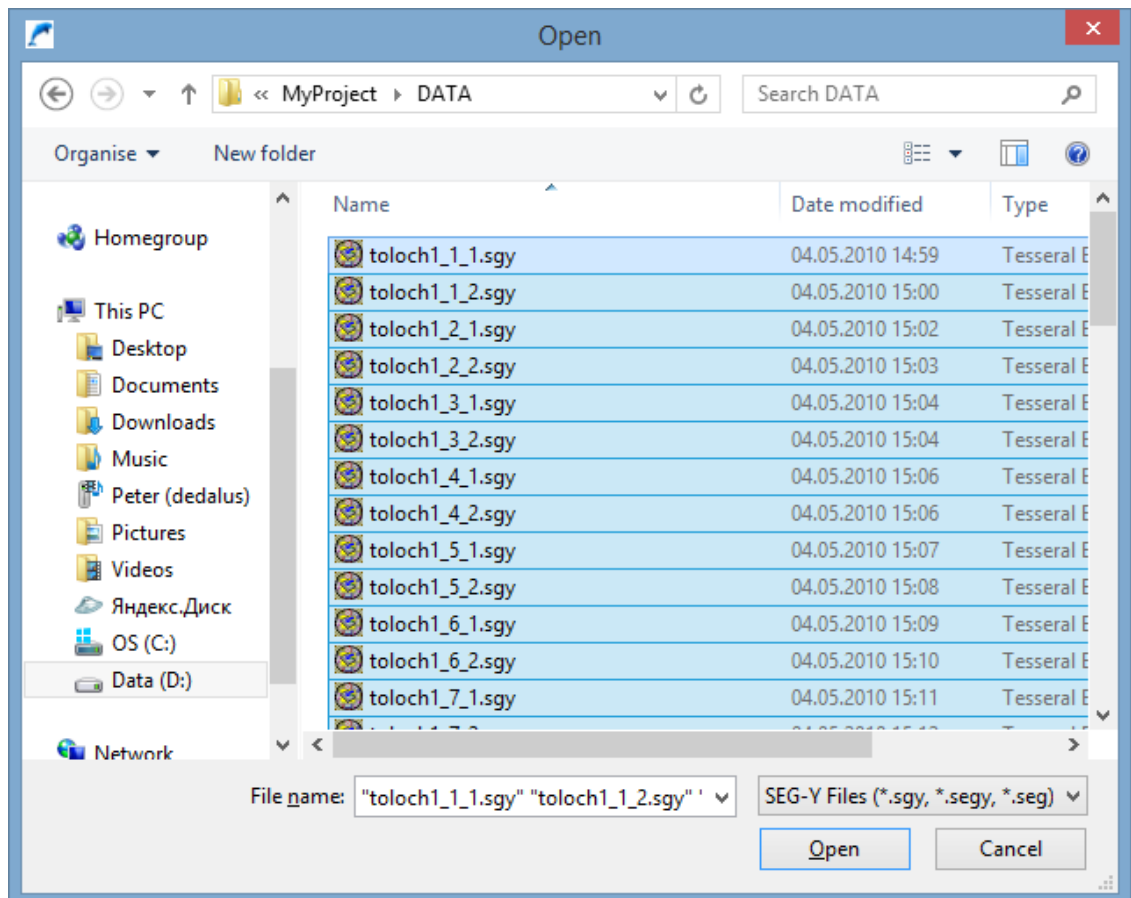

On the right here is the list of available processing modules divided into groups depending on their function: Data I/O, Geometry/Headers, Interactive Tools, Signal Processing, etc. On the left is the flow itself, which by now is empty. We are going to populate the flow with the processing modules and then execute it.

10. Our data input flow shall read the data from an input file and save it as a dataset object into the project database. Since our data is in SEG-Y format, so to read it we will use *SEG-Y Input* module. Catch it in the module list on the right with the left mouse button, drag to the left part of the window and drop within the processing flow area. You will see the module parameter dialog:
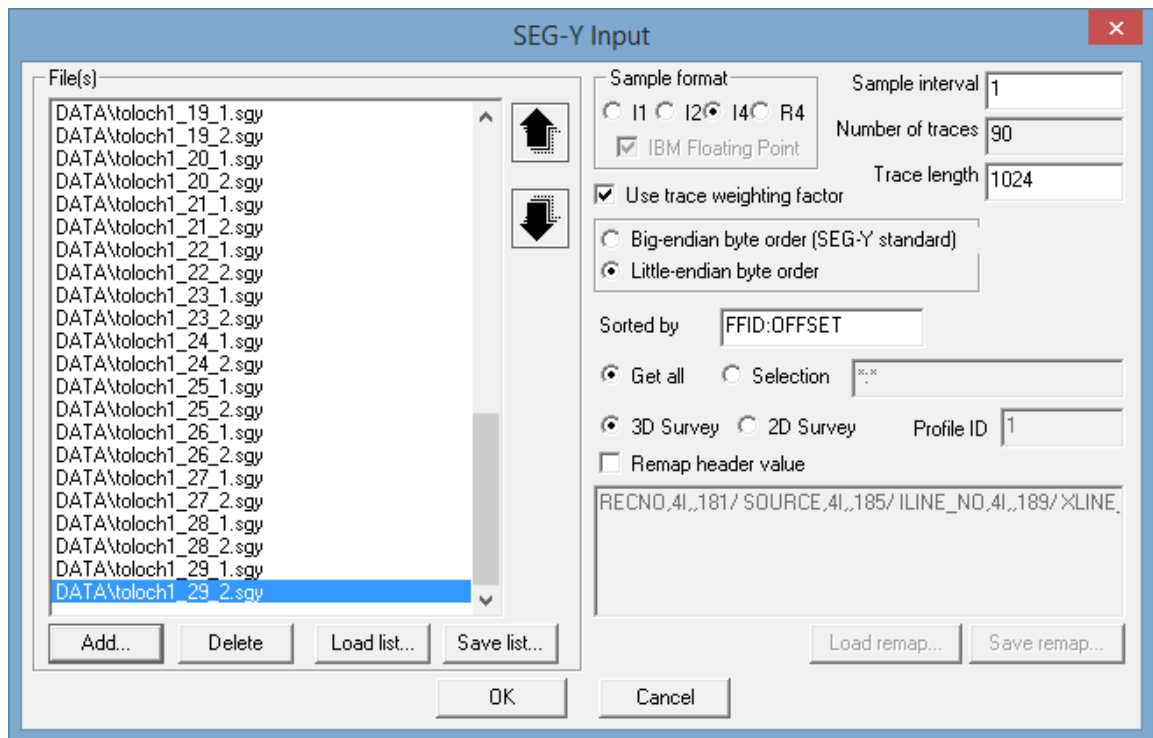
---

[1] If you are loading the data to an existing project, you may wish to create a new *line* and a new data input *flow* for it, see step 7 (b, c).

Click the Add… button to select files. Get into the DATA folder and press Ctrl+A to select all SEG-Y files there at once.
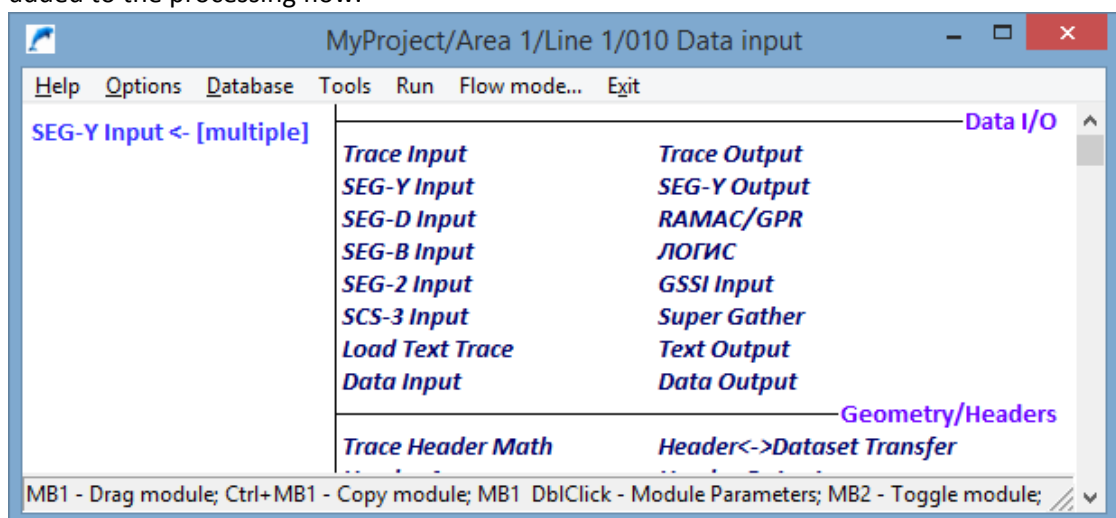

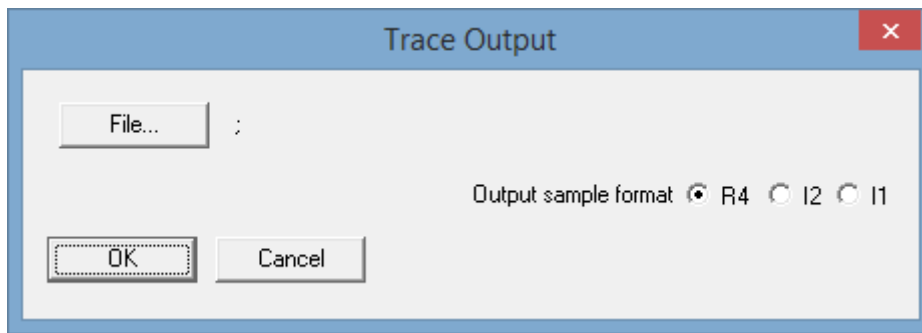Click the Open button to add all of them into the list:

You can scroll through the list to make sure that all the files are there and the order is correct. If the order is not correct, you may use the up and down arrow buttons to the right of the list to move a selected file through the list.

The module detects automatically the parameters of the SEG-Y files. As you can see from the picture, here they were correctly detected as 4-byte integer sample format, little-endian byte order, 1 ms sampling interval, trace length – 1024 samples. If, in your case, any of these parameters were detected wrongly, you may always correct those manually indicating correct values.

After you have finished with the SEG-Y Input parameters click the OK button to have the module added to the processing flow.
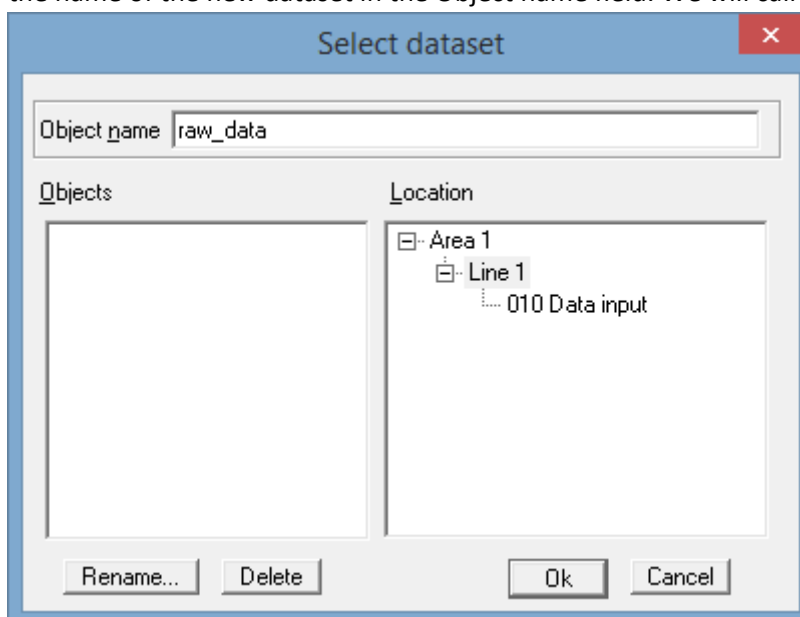

11. The data is saved to the project database using the *Trace Output* module. Add it to the flow the same way as you did it with the SEG-Y Input, by drag-and-drop. You will see its parameter dialog:

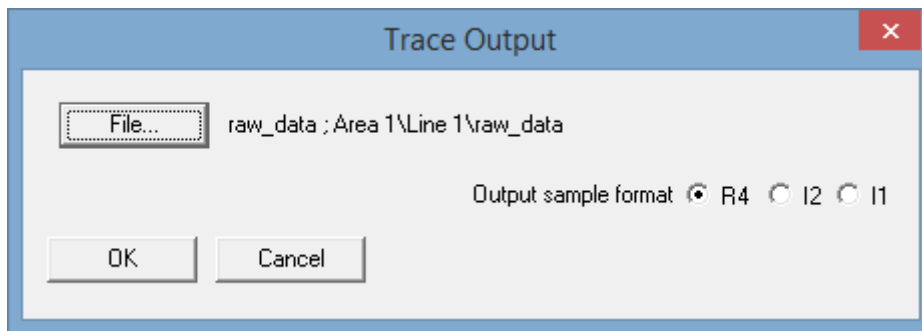Click the File… button to see the project database browser dialog:

Here on the right you see the project database structure (with the area, line and flow we have created). Click on the level of the database where you wish to save your data (we normally recommend that the data is saved on the line-level, however this is not obligatory). Then, type the name of the new dataset in the Object name field. We will call this dataset 'raw_data':
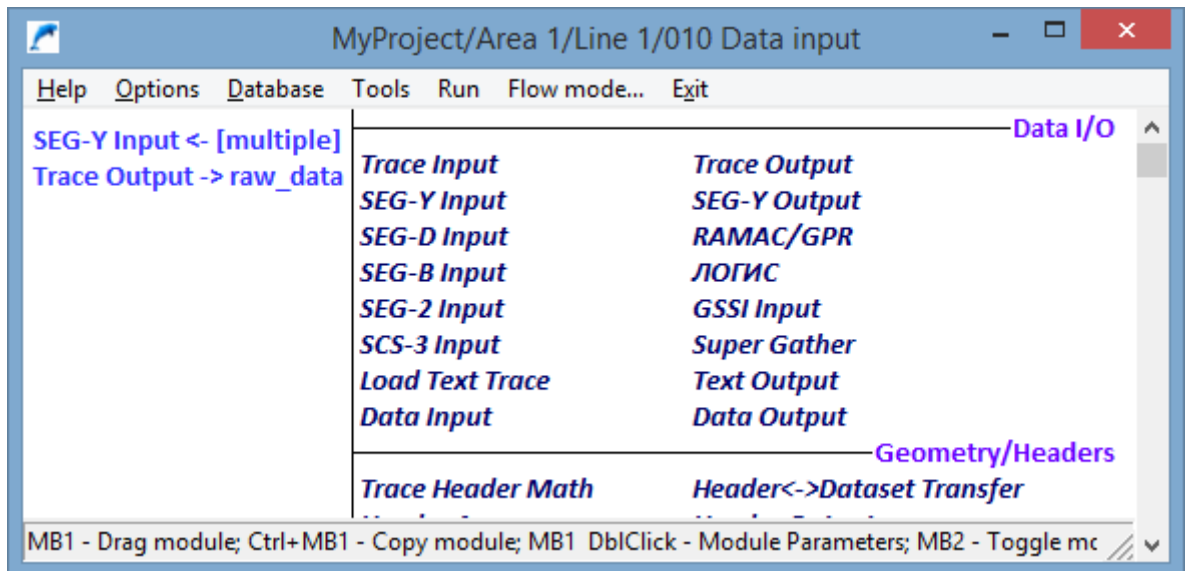

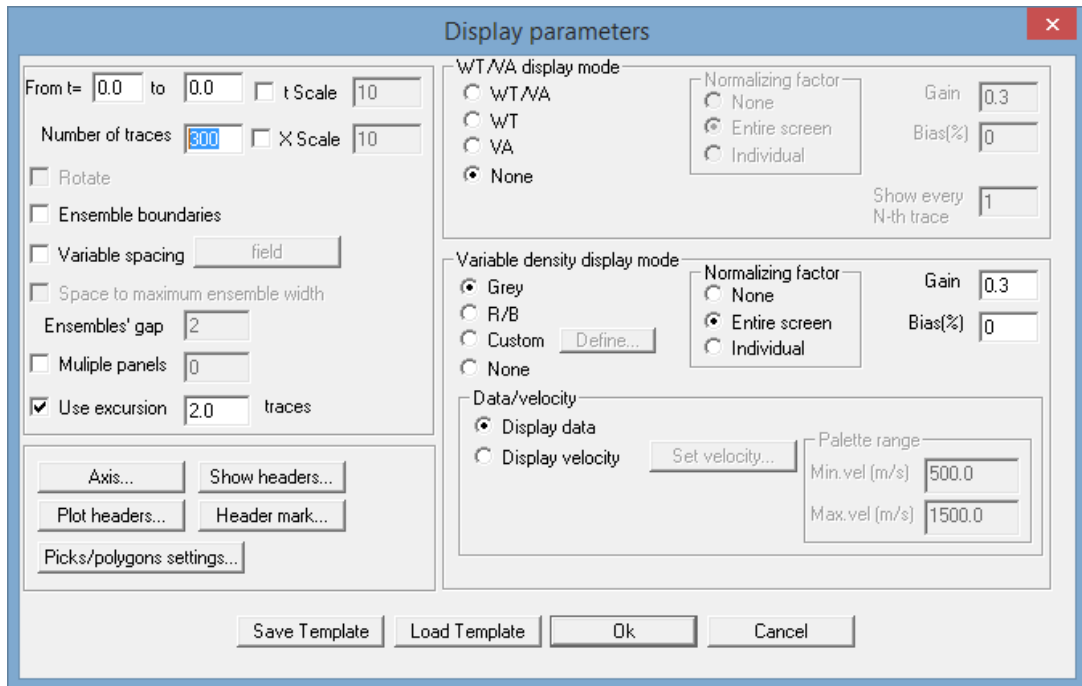Click the OK button. Now the path to the new dataset is shown in the Trace Output dialog:
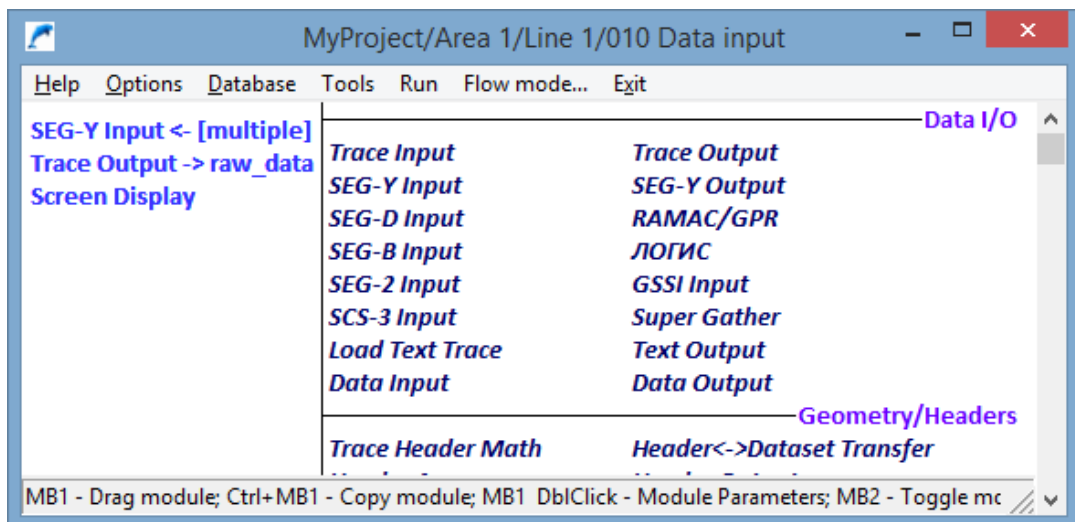
Click the OK to save the module to the flow.

12. Now, we have a flow which, when executed, will read all our shots from multiple files and save them all into one dataset called 'raw_data'. This is enough to load the data, however, we would like to see what we are loading on the screen[2]. To have this, we will add a module called *Screen Display* to the end of the flow.

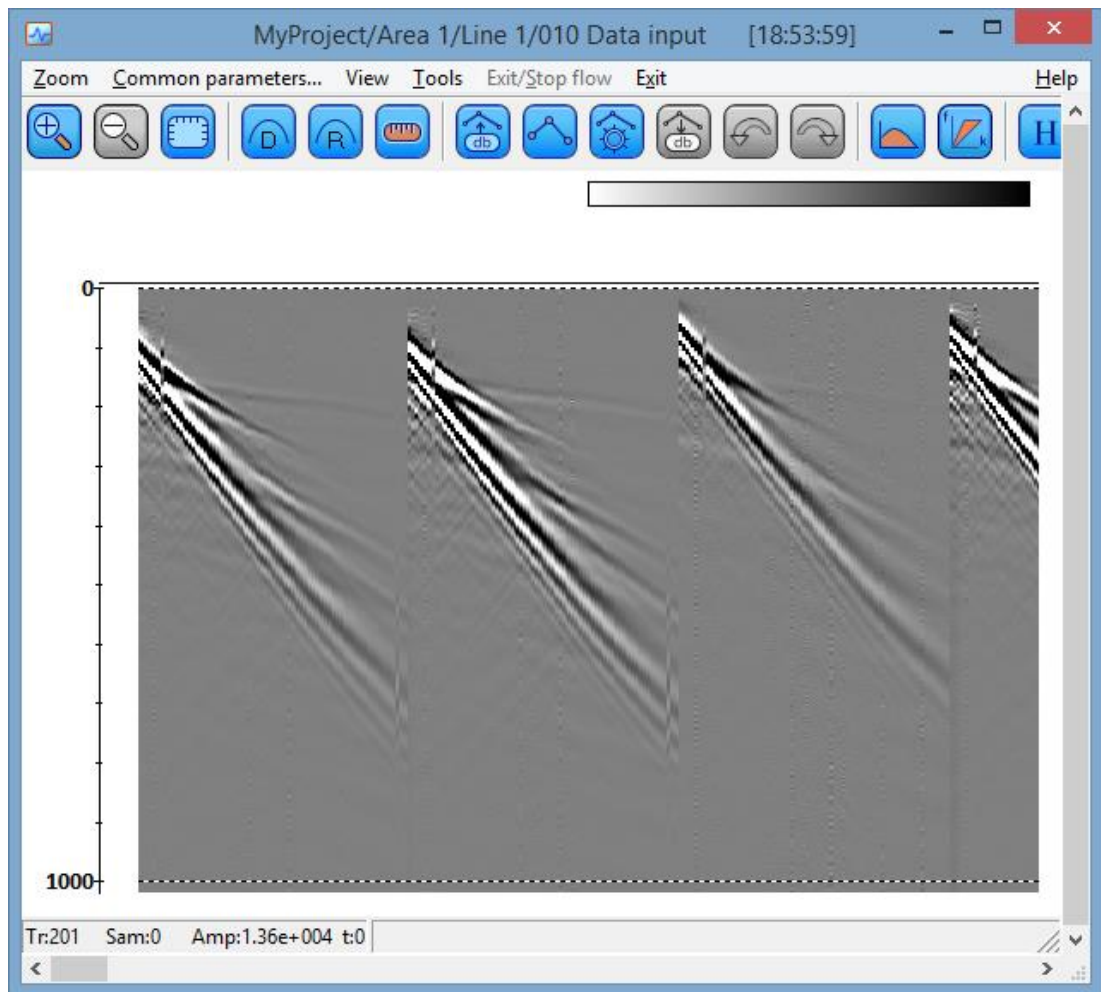You may use the default parameters of the Screen:

---

[2] If the amount of data you are loading is fairy big (about 1 GB and more), it could be more efficient to use *framed mode* of flow execution to load it frame by frame (framed mode is available through the Flow mode… entry of the main menu). In this case, you probably would not like to display the data on the screen because it will pause your flow at the end of each frame – skip the step 12 then.

Click the OK to see the flow as shown below:

13. Our flow is ready. Click the Run command of the flow editor main menu to execute the flow. When executed it will read the data from SEG-Y files, save it to the 'raw_data' dataset and finally display it on the screen. You will see the display similar to what is shown below:

14. Now, when the data is loaded into the project database, you may load it to any other flow using the *Trace Input* module.